IN THE CLAIMS

Please amend the claims as indicated:

1. (currently amended)   A method for language verification of a [[Java card CAP]] reduced file to be stored in and used by a chipcard, wherein the reduced file is created from an original [[Java]] code file, the method comprising:

a) a conversion step for converting said [[Java card CAP]] reduced file into a corresponding converted [[Java]] reduced code file that is semantically identical to said [[Java card CAP]] reduced file, wherein the [[Java card CAP]] reduced file is created from an [[the]] original code file that contains classes that are capable of being compiled, and wherein the only executable instructions in the [[Java card CAP]] reduced file are applets, wherein said conversion step further includes:

a preconversion substep for converting [[Java]] card IDs contained in said [[Java card CAP]] reduced file into symbolic names, and for converting said [[Java card CAP]] reduced file into a standard [[Java]] language format used by a condensed interpreted language, to obtain a preconverted file; and

a mapping substep for replacing, in said preconverted file, externally defined names with original names by using a mapping scheme between [[Java]] names used by the reduced file and tokenized identifiers, to obtain the converted [[Java]] reduced code file for a language-verification step; and

b) a language-verification step for verifying said converted [[Java]] reduced code file for compliance with [[Java]] pre-specified language specifications of the condensed interpreted language.

2. (cancelled)

3. (currently amended)   The method for language verification of [[a Java card CAP]] the reduced file according to Claim 1, wherein said mapping substep is performed using a referenced [[Java]] export file which is available as a result of creating said [[Java card CAP]] reduced file from said original [[Java]] code file.

4. (currently amended)   The method for language verification of [[a Java card CAP]] the reduced file according to Claim 1, the method further comprising:

c) a signature step for creating, after verification of said converted [[Java]] reduced code file in said language verification step, a cryptographic signature file for the [[Java card CAP]] reduced file.

5. (currently amended)   The method for language verification of [[a Java card CAP]] the reduced file according to Claim 4, further comprising:

d) a loading step for loading the cryptographic signature file to a chipcard together with the [[Java card CAP]] reduced file, wherein the cryptographic signature file is attached to the [[Java card CAP]] reduced file when loaded in the chipcard.

6. (currently amended)   The method for language verification of [[a Java card CAP]] the reduced file according to Claim 4, wherein the cryptographic signature file is cryptographically verifiable, said method further comprising:

e) an executing step for executing said [[Java card CAP]] reduced file upon a positive crypotgraphic verification.

7-9. (cancelled)

10. (currently amended)   A tangible computer-readable medium [[embodying]] storing computer program code, the computer program code comprising computer executable instructions configured for language verification of a reduced file that is created from an original code file, wherein the reduced file is to be stored in a chipcard, and wherein the computer executable instructions, when executed, perform the steps of:

converting [[said Java card CAP]] said reduced file into a corresponding converted [[Java]] reduced code file that is semantically identical to said [[Java card CAP]] reduced file;

verifying said converted [[Java]] reduced code file for compliance with [[Java]] pre-specified language specifications;

converting [[Java]] card IDs contained in said [[Java card CAP]] <u>reduced</u> file into symbolic names;

converting said [[Java card CAP]] <u>reduced</u> file into a standard [[Java]] format <u>of a condensed interpreted language,</u> to obtain a preconverted file; and

replacing in said preconverted file externally defined names with original names by using a mapping scheme between [[Java]] names <u>used by the reduced file</u> and tokenized identifiers, to obtain the converted [[Java]] <u>reduced</u> code file.

11. (cancelled)

12. (currently amended) A <u>tangible</u> computer-readable medium [[containing]] <u>storing</u> computer program code for a [[Java card CAP]] <u>reduced</u> file language verifier for verifying a [[Java card CAP]] <u>reduced</u> file that has been derived from an original [[Java]] code file <u>written in a condensed interpreted language,</u> said [[Java card CAP]] <u>reduced</u> file including original [[Java]] <u>pre-specified language</u> semantics of said original [[Java card]] <u>code</u> file, the computer program code comprising instructions for:

a converter for converting said [[Java card CAP]] <u>reduced</u> file into a corresponding converted [[Java]] <u>reduced</u> code file that is semantically identical to said [[Java card CAP]] <u>reduced</u> file, wherein said converter further includes:

a preconverter for converting [[Java]] card IDs contained in said [[Java card CAP]] <u>reduced</u> file into symbolic names, and for converting said [[Java card CAP]] <u>reduced</u> file into a standard [[Java]] <u>condensed interpreted language</u> format, to obtain a preconverted file; and

a mapper for replacing in said preconverted file externally defined names with original names under use of a mapping scheme, to obtain the converted [[Java]] <u>reduced</u> code file; and

a language verifier for verifying said converted [[Java]] <u>reduced</u> code file upon its compliance with a [[Java]] language specification <u>of the standard condensed interpreted language.</u>

13. (cancelled)

14. (currently amended) The <u>tangible</u> computer-readable medium according to Claim 12, wherein the mapper comprises an input for receiving a referenced [[Java]] export file created when a referenced [[Java]] card [[CAP]] file was converted from its corresponding original [[Java]] code file.

15. (currently amended) The <u>tangible</u> computer-readable medium of Claim 12, wherein the instructions are further configured for a signature generator for generating a second cryptographic signature file.

16. (currently amended) The <u>tangible</u> computer-readable medium of Claim 15, wherein the instructions are further configured for loading the second cryptographic signature file and the [[Java]] card [[CAP]] file to a storage device.

17. (currently amended) A <u>tangible</u> computer-readable medium [[containing]] <u>storing</u> computer program code for a reduced file language verifier for verifying a reduced file that has been converted from an original file, the reduced file maintaining original semantics of the original file, the computer program code comprising instructions for:

a converter for converting said reduced file into a corresponding converted file that is semantically identical to said reduced file, wherein said converter further includes:

a preconverter for converting IDs contained in said reduced file into symbolic names and for converting said reduced file into a standard format, to obtain a preconverted file; [[and]]

a mapper for replacing in said preconverted file externally defined names with original names under use of a mapping scheme, to obtain the converted file;

means for determining whether said reduced file complies with a predetermined language specification; and

a language verifier for verifying said converted file upon compliance with the predetermined language specification.

18. (cancelled)

19. (currently amended)   The <u>tangible</u> computer-readable medium of Claim 17, wherein said mapper comprises an input for a referenced difference file which is available as a result from a conversion in which a referenced reduced file has been converted from its original file.

20. (currently amended)   A method comprising:

    converting an original file into a reduced file, wherein the original file contains a class description section and an instruction section, and wherein the reduced file contains a code description section that is based on the class description section, and wherein the reduced file contains a code section that is based on the instruction section, wherein the original file contains classes that are capable of being compiled, and wherein the only executable instructions in the reduced file are applets;

    converting the reduced file into a converted file, wherein the reduced file and the converted file are semantically identical<u>, wherein the converted file is created by a preconversion step and a mapping step, wherein the preconversion step converts identifier names in the reduced file into symbolic names to generate a preconverted file that comprises a preconverted code section and a preconverted code description section, and wherein the mapping step replaces the symbolic names with original names used by the original file to create the converted file</u>;

    creating a cryptographic signature for the converted file; and

    storing the cryptographic signature and the reduced file in a chipcard, wherein the cryptographic signature verifies that the reduced file was converted by a trusted entity.

21. (cancelled)

22. (currently amended)   The <u>tangible</u> computer-readable medium of claim 10, wherein the [[Java card CAP]] <u>reduced</u> file is created from an original file that contains classes that are capable of being compiled, and wherein the only executable instructions in the [[Java card CAP]] <u>reduced</u> file are applets.